



Videojuegos Retro: creatividad e ingenio para crear una consola desde cero

Antoni Perez-Poch

Dept. Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya · BarcelonaTech (UPC)

EUETIB Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona

antoni.perez-poch@upc.edu

Resumen

En este artículo se describe la introducción de una propuesta de trabajo innovadora para los alumnos de Ingeniería, Grado o Máster, con intensificación en electrónica o informática. La propuesta se realiza como síntesis final de la titulación en el Trabajo Fin de Grado, pero puede realizarse también como parte de alguna asignatura de la titulación.

Los alumnos fabrican su propia consola de videojuegos retro a partir de unas especificaciones iniciales y un microcontrolador programable. El trabajo incluye el diseño de *hardware* electrónico, programación en lenguaje ensamblador o de alto nivel de los videojuegos implementados, así como fabricación y comprobación del prototipo. La característica lúdica del proyecto es altamente motivadora. Además, la creatividad tiene un rol importante, ya que el alumno diseñará tanto el aspecto final de la consola como el propio juego, basándose en las consolas pioneras de los años setenta y ochenta. El concepto de un videojuego retro de gráficos sencillos, pero totalmente jugable, permite que la dificultad del trabajo sea razonable para su realización en un curso académico. Llevamos realizando esta experiencia desde hace cinco años en la EUETIB, Escuela Universitaria de Ingeniería Técnica Industrial de Barcelona, centro asociado a la UPC, Universitat Politècnica de Catalunya (BarcelonaTech), con resultados altamente satisfactorios.

Palabras clave: Creatividad, videojuegos, programación, diseño electrónico, trabajo fin de Grado.

Recibido: 20 de marzo de 2014; **Aceptado:** 14 de abril de 2014.

1. Introducción

Dentro de las titulaciones de Grado en Ingeniería Industrial, el Trabajo Fin de Grado (TFG) está planteado con un objetivo finalista. Su objetivo en la EUETIB es ofrecer la oportunidad al alumno de poner en práctica los conocimientos y destrezas adquiridos durante la titulación.

En nuestro caso, nos centramos en proyectos dirigidos a alumnos del Grado en Electrónica y Automática. Dado que el número de horas a dedicar es limitado por parte del alumno (equivalente a 24 créditos), es necesario establecer un compromiso entre las horas dedicadas a programación, diseño y realización del prototipo *hardware*.

Los videojuegos son un campo especialmente motivador para los alumnos. Sin embargo, la alta especialización gráfica y la complejidad *hardware* de los sistemas de videojuegos actuales hacen que sea difícil que un alumno pueda combinar en su trabajo las dos partes mencionadas, *software* y *hardware*. En cambio, los sistemas de videojuegos denominados “retro” (los que aparecieron en los años setenta y ochenta) presentan menor complejidad de desarrollo sin perder por ello su fondo de motivación del alumno.

Actualmente, los microcontroladores ofrecen la posibilidad de recortar el tiempo de desarrollo *hardware* al integrar gran parte de la complejidad del que sería el circuito electrónico equivalente en un programa realizado en ensamblador. Un ejemplo de ello es el videojuego Pong, un clásico de la era analógica de los videojuegos que consistía en un juego de ping-pong primitivo que utilizaba la propia televisión. Si bien la complejidad del circuito electrónico que forma una consola original de este tipo es elevada, esta complejidad se reduce si el circuito se implementa con un microcontrolador electrónico [2].

Además, esta opción de diseño de *hardware* tiene la ventaja de su flexibilidad con respecto al prototipo con componentes discretos. En efecto, es posible alterar el juego ofreciendo nuevas y variadas opciones, o incluso proponer juegos totalmente nuevos o basados en otros clásicos de la época. Con el uso del microcontrolador se puede dar opción a la creatividad del alumno y que además se ejercite en la implementación de programas en un lenguaje específico que no ha sido cubierto antes en otras asignaturas.

En los siguientes apartados se describen tres prototipos de

proyectos piloto de esta estrategia [4, 7, 8], con detalles de su diseño e implementación para que pueda ser reproducible por parte de los profesores y alumnos interesados en la materia.

El montaje desarrollado está formado por un *hardware* y un *software*. El *hardware* permite crear un vínculo con el exterior, permitiendo que los jugadores interaccionen con él. Todo este *hardware* se controla mediante un *software* grabado en el microcontrolador. De esta manera se consigue que la unión de estos dos elementos funcione, obteniendo así una videoconsola moderna y totalmente jugable, pero con aspectos retro.

En el siguiente apartado describimos una propuesta de metodología docente de trabajo con los alumnos de Trabajo Fin de Grado en Ingeniería Industrial, dividida en anteproyecto y proyecto final. Además se describen otras propuestas didácticas para otras asignaturas del temario. Posteriormente, en las secciones 3,4 y 5 se detallan ejemplos de prototipos realizados con esta metodología, de dificultad creciente. Finalmente, se explicitan las conclusiones del artículo así como algunas líneas de futuro trabajo.

2. Metodología de trabajo

¿Cómo empezar a trabajar en nuestra docencia con los videojuegos retro? La primera decisión consiste en plantearse qué objetivos didácticos se pretenden.

¿Deseamos realizar un Trabajo Fin de Grado o Máster, o una práctica de una asignatura? ¿O incluso basar toda la metodología de una asignatura en la realización de un proyecto, como se hace con la técnica *Project Based Learning* (PBL) [3]?

El proyecto que se muestra en este artículo es de suficiente envergadura como para realizar dos proyectos fin de carrera enteros. Se trata de un trabajo que debe potenciar el trabajo autónomo del alumno y sintetizar los conocimientos adquiridos durante la titulación.

En nuestro caso, dividimos la realización de un Trabajo Fin de Grado o Máster en dos partes: anteproyecto y el proyecto en sí. En su totalidad, el trabajo realizado equivaldrá a una dedicación global de 24 créditos.

2.1. Anteproyecto

Desde la fecha de registro del proyecto hasta su matrícula efectiva, el alumno debe realizar el anteproyecto. Para ello, empleará aproximadamente seis semanas lectivas. Algunos de los objetivos que debe cubrir el alumno son:

- Haber definido los objetivos generales y específicos de su proyecto.
- Haber planteado diversas alternativas técnicas y justificar la elección de una de ellas como la más idónea.
- Describir la ingeniería de concepción de su futuro proyecto: ¿qué diseño va a tener? ¿con qué metodología de trabajo va a realizarse? ¿cuál es el esquema de bloques del diseño final?

- Realizar un cronograma con la planificación detallada de su propio trabajo en la segunda fase.
- Realizar un análisis de puntos críticos e hitos del proyecto, mediante un diagrama de Gantt o similar.
- Prever los recursos necesarios, tanto técnicos como económicos.

En resumen, se trata de realizar un “Anteproyecto llave en mano”. El anteproyecto debe ser lo suficientemente explicativo y detallado para que cualquier otro proyectista fuese capaz de desarrollar el proyecto entero.

Por supuesto, en un proyecto como este, donde se unen programación y diseño de *hardware*, es necesario realizar un primer prototipo. En el anteproyecto se pide, pues, que el alumno demuestre que tiene unas mínimas destrezas con las herramientas de desarrollo que va a utilizar. ¿Conoce la familia de microcontroladores que va a usar? ¿Sería capaz de hacer un programa en bajo nivel sencillo, compilarlo y ejecutarlo? ¿Dispone del banco de desarrollo y sabe cómo hacerlo funcionar?

En nuestro caso, optamos por una tutorización del proyecto presencial y semanal, de manera que se lleva un registro de los avances del alumno a lo largo del curso. En él se registra si el alumno finaliza esta primera parte, de manera que se considere viable finalizar el proyecto en un cuatrimestre.

Creemos que el proyecto debe ser en primer lugar, bien dimensionado. Ni demasiado fácil ni demasiado extenso. La mayoría de consolas de videojuegos actuales son extraordinariamente complejas. Queda fuera de lugar realizar una videoconsola como un proyecto fin de carrera.

Sin embargo, los videojuegos retro como el proyecto que presentamos en este artículo son suficientemente sencillos para poder incluir ambas cosas: la programación del videojuego y el diseño electrónico de la propia consola. A este efecto, los microcontroladores son de gran utilidad, ya que reducen la complicación del diseño de *hardware* y permiten evaluar la destreza de programación a bajo nivel del alumno.

Concretamente el alumno deberá realizar las siguientes tareas:

- El diseño del programa-juego a realizar.
- La realización de rutinas básicas.
- Las pruebas con el simulador de programación del microcontrolador y el entrenamiento con el kit de desarrollo.
- El análisis y la elección del microcontrolador óptimo para el programa escogido.
- El análisis y la elección del visualizador a utilizar: LED individuales, matriz de LED, etc.
- El diseño de bloques del circuito electrónico que formará la consola.
- El diseño inicial de la placa de circuito a impreso, sin optimizar.

2.2. Realización del Trabajo Fin de Grado

Abordamos ahora la tarea de tutorizar al alumno hasta la finalización del videojuego. Como hemos comentado, el punto crítico creemos que es el de dimensionar correctamente el proyecto de manera que sea factible realizarlo por parte del alumno. Esta tarea ya está hecha en este momento. Ahora el alumno debe ejecutar el anteproyecto que él mismo ha presentado.

Como en todo proyecto, pueden salir imprevistos, retrasos y problemas técnicos que retrasen su desarrollo. El profesor tutor tendrá un doble rol, por un lado el del cliente final, que quiere que su trabajo esté acabado en la fecha prevista y con unos mínimos criterios de calidad. Por otro lado, el profesor tutor estará atento a hacer recomendaciones al alumno y orientarlo, es decir hará de gestor del proyecto por encima de la planificación acordada.

En la evaluación del TFG, como es tradicional, hay un tribunal, que en nuestra Escuela está formado por dos profesores de la propia área de conocimiento y un profesor de otra área de conocimiento distinta. El profesor tutor no forma parte del tribunal, pero es práctica habitual que su opinión sea escuchada por el tribunal y tenida en consideración. Por todo ello, continuamos con la política de evaluar periódica y presencialmente al alumno, una vez por semana, según los objetivos parciales establecidos previamente.

Además, el alumno ha de presentar una memoria que es estándarizada según unas pautas publicadas por la escuela. Como mínimo, la memoria debe contener:

- Los apartados del anteproyecto que sean relevantes para el producto final.
- La descripción de la ingeniería de detalle: qué componentes se han utilizado, cuál es el circuito utilizado, qué subprogramas forman parte del código y cuál es su estructura.
- Un apartado de bibliografía y conclusiones.
- Los anexos con los planos del circuito impreso, las hojas de referencia de los componentes y el código utilizado.
- Los aspectos legales y las normativas de aplicación.

En el caso que nos ocupa es importantísimo tener el prototipo *hardware* realizado cuanto antes. De esta manera la tarea de hacer pruebas con el programa del microcontrolador se simplifica. Algunos microcontroladores disponen de un kit de desarrollo con simulador, pero muchas veces, al pasar el diseño al prototipo real, aparecen nuevos problemas técnicos que hay que abordar.

No hay que olvidar las pruebas o comprobaciones de funcionamiento y usabilidad, que deben ser debidamente diseñadas y detalladas en la memoria. El alumno, al final, ha de ser capaz de describir el ciclo entero de vida del proyecto, desde su concepción a partir de los requisitos de un jugador de la

videoconsola, hasta su prueba real con la realimentación correspondiente.

La memoria final debe incluir también una guía del usuario del videojuego y un presupuesto completo.

La evaluación final se basa en la defensa tradicional del TFG mediante una exposición oral y preguntas, junto con una demostración práctica del elemento *hardware* implementado.

2.3. Otras propuestas didácticas

A pesar de que nuestra experiencia se basa en un Trabajo Fin de Grado, creemos, como hemos comentado anteriormente, que también se puede adaptar a parte o una asignatura entera.

En este sentido, se puede programar como un proyecto por partes para una asignatura de diseño de *hardware* electrónico. La realización de estas prácticas pueden plantearse a partir de guiones realizados *ad hoc*, de manera que los alumnos vayan construyendo incrementalmente la placa, y posteriormente se puede programar también el microcontrolador.

También puede ser adecuado para una asignatura de informática industrial o de programación de microcontroladores. Proponemos aquí que la placa del videojuego se entregue ya realizada a los alumnos y que éstos se centren en las tareas de programación.

Finalmente, aquí se ha planteado el proyecto como algo de realización individual, pero estimamos que tiene suficiente entidad para como realizarlo en parejas, o bien en grupos mediante metodología cooperativa. Por ejemplo, cada grupo podría programar un juego determinado y realizar al final una competición entre ellos.

Una opción que parece muy razonable para una asignatura de últimos cuatrimestres es basarla en un proyecto como este, de manera que la evaluación de toda la asignatura se base en la implicación de varios alumnos. Para ello el profesor debe establecer objetivos parciales, mecanismos periódicos de evaluación y autoevaluación formativa e interacciones de cooperación. Se trata de la metodología PBL [3], que cada vez podemos encontrar más a menudo en nuestras universidades.

A continuación, describimos sintéticamente en los apartados 3, 4 y 5 tres ejemplos de desarrollos con sus correspondientes referencias [4, 7, 8]. En la bibliografía se pueden encontrar referencias a otros proyectos similares desarrollados en nuestro laboratorio [5, 10]. Algunos de los ejemplos descritos [4, 5, 7] corresponden a proyectos fin de carrera de Ingeniería Técnica Industrial realizados según el plan de estudios pre-Bolonia, aunque las conclusiones presentadas y el tiempo de realización son equivalentes.

3. Descripción de una experiencia práctica: videoconsola básica

Presentamos a continuación la descripción del proyecto de creación de una videoconsola retro básica [7].

La Figura 1 muestra un diagrama de bloques de los elementos generales de los que consiste todo el conjunto del *hardware* de nuestro prototipo. En el *hardware* se distinguen dos partes claramente determinadas por las funciones que realizan:

- La parte de *control* que se encarga, como su nombre indica, de controlar todos los elementos presentes en el circuito.
- El circuito que se encarga de la *visualización* del videojuego, que está formado por cuatro matrices de LED de cinco filas por siete columnas.

3.1. Hardware

El elemento mas importante del esquema de la placa de control es el microcontrolador PIC 18F452 de Microchip, que con sus 32 Kbytes de memoria de programa, y sus entradas analógicas de 10 bits, es el encargado de albergar y ejecutar el programa que controla el resto del *hardware* para hacer posible el funcionamiento de esta videoconsola.

Una parte también importante del circuito es la que permite interactuar a los jugadores con el juego. Por un lado se encuentran los *joysticks* que permiten a los jugadores mover las raquetas por el campo así como también seleccionar el juego deseado. Y por otra parte se encuentran los diferentes interruptores y pulsadores con los que se pueden controlar diversos aspectos de los juegos como la velocidad de juego, la selección de los juegos, selección de dificultad y reinicio de la consola.

Para el desarrollo se han usado las técnicas habituales en el diseño de placas de microcontroladores y lenguaje ensamblador descritas en la bibliografía [1, 6, 9].

Los elementos que se encuentran en la placa de control son componentes estándar, como por ejemplo el regulador 7805 que se encarga de asegurar que la alimentación del circuito sea de 5 V casi uniformes. También se encuentran los interruptores y pulsadores con los que el usuario puede interactuar con el juego. Están conectados de tal manera que mientras el contacto esté abierto el micro, reciba un nivel alto, y al cerrar dicho contacto reciba un nivel bajo.

En la placa donde se encuentra la pantalla por donde se mostrarán los juegos intervienen pocos elementos, haciendo bastante sencillo su montaje. Cada matriz solo dispone de una conexión para cada una de sus filas enteras y otra para cada columna entera, pudiendo controlar sus LED con coordenadas.

Para poder visualizar la puntuación de los jugadores en cada uno de los juegos se han utilizado 2 visualizadores de 7 segmentos de color rojo.

3.2. Programación y desarrollo del *software*

El *software* desarrollado se ha escrito en lenguaje ensamblador para optimizar el código. El programa completo consta

de 3 juegos míticos en los inicios del mundo de las videoconsolas. Entre ellos encontramos los conocidísimos Pong, Fútbol y Arkanoid. Al encender la consola lo primero que aparece por pantalla es el menú de selección de juego. La Figura 2 muestra un ejemplo del menú de selección.

Cuando el jugador selecciona el juego del Pong el *software* inicia una serie de rutinas iniciales de control de variables, donde da valores determinados para poder ejecutar correctamente el juego. En este juego las interacciones posibles son las siguientes:

- 2 jugadores simultáneos. El juego está diseñado para que jueguen 2 personas, una contra la otra.
- Opción de 2 velocidades de juego. Se permite elegir la velocidad lenta o rápida.
- Posibilidad de reinicializar la consola. Se permite finalizar el juego cuando se desee e ir al menú de selección.
- Saque de la bola. Permite realizar el saque de la bola.

La Figura 3 muestra la consola ejecutando el juego del Pong.

En el caso de que el jugador seleccione el juego del Fútbol, el puntero del programa saltará a la posición donde se encuentra el código del juego. Este juego es bastante parecido al Pong, ya que mantiene las mismas opciones de interacción con el usuario.

La gran diferencia es que se ha dotado a cada jugador de unas raquetas extras, simulando a delanteros de fútbol. Ahora cada jugador, aparte de disponer de la raqueta de portero que mantiene el aspecto de las del Pong, dispone de una raqueta delantera. Aunque parezca un juego muy parecido al Pong, proporciona una jugabilidad más entretenida y difícil de controlar, haciéndolo más ameno. La Figura 4 muestra el campo de juego del Fútbol.

El tercer y último juego consiste en el Arkanoid. Se trata de un juego en el que mediante una bola se ha de ir derribando los distintos ladrillos que conforman las estructuras de cada pantalla. La finalidad del juego es derribar todos los ladrillos de todas las pantallas evitando perder vidas. El juego finaliza al llegar a los 99 puntos. La bola se controla mediante una plataforma similar a la de los otros juegos.

Las interacciones entre el usuario y la consola cambian ligeramente respecto a los otros juegos:

- Juego para un solo jugador. El juego solo dispone de un joystick.
- Botón de saque. Opción igual para todos los juegos.
- Pulsador de reinicio.
- Velocidad rápida y lenta.
- Dificultad fácil o difícil. Esta opción es sólo para el Arkanoid. La pala del jugador se hace más pequeña.

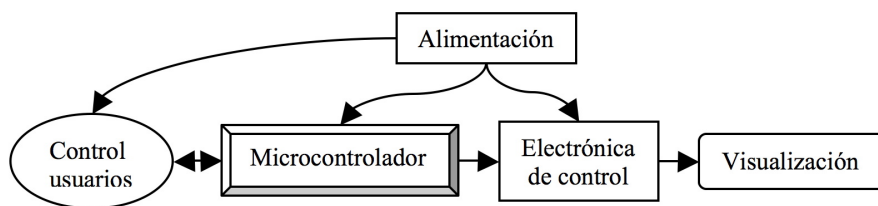


Figura 1: Diagrama de bloques del sistema videojuego retro. (Crédito: A. Moser).



Figura 2: Selección del juego Pong. (Crédito A. Moser).



Figura 3: Juego Pong en funcionamiento. (Crédito: A. Moser).



Figura 4: Juego Fútbol. (Crédito: A. Moser).

En este juego la estructura sigue siendo la misma, aunque las instrucciones son totalmente distintas. El programa tiene que cargar cada pantalla cogiendo los datos de una tabla. La parte más complicada y destacada del juego es el control de choque, ya que en este caso tiene que mirar si choca con algún ladrillo dependiendo de la dirección y sentido de la bola. Se puede decir que de los 3 juegos éste es el más complejo, debido a la interacción que se ha de conseguir con las diferentes pantallas. En la Figura 5 se puede ver una pantalla del juego.

Como vemos, la propuesta presentada es suficiente flexible para complicar más o menos la tarea tanto del diseño de *hardware* como de programación. Existen numerosos juegos básicos que pueden también implementarse. El profesor podrá adaptar esta propuesta al nivel y ámbito docente que corresponda, y podrá además ofrecer retos de creatividad a los alumnos proponiendo que diseñen nuevos juegos con los que entretener a sus amigos y amigos.

4. Descripción de una experiencia práctica: videoconsola con visualizador matricial

En esta segunda experiencia se realiza un proyecto basado en la experiencia anterior, pero aumentando las prestaciones del *hardware* y programando videojuegos diferentes [4].

En esta implementación se proponen videojuegos para un solo jugador, aunque la videoconsola estará preparada para su uso por dos jugadores.

4.1. Hardware

Las principales diferencias de este proyecto con respecto al anterior se encuentran en la placa destinada a la visualización (Figura 6). En ella se ha aumentado el número de matrices de 4 a 9 obteniendo una pantalla de 315 LED o píxeles en total (siendo las matrices de 5 por 7 LED). El número de visualizadores se ha incrementado en 1 dígito, pudiendo visualizar puntuaciones de hasta 3 dígitos. El aumento del número de píxeles tiene como efecto aumentar la jugabilidad.

El control de las matrices, como puede verse en la Figura 7, es parecido a la escritura de una memoria externa: se selecciona una dirección o columna y se le escribe un dato, es decir, qué LED están encendidos y cuáles no en esa columna. Para conseguir este tipo de control se han conectado las filas de las matrices directamente al microcontrolador y las columnas a un decodificador comandado por el microcontrolador.

Para la conexión de los visualizadores se ha aprovechado que el número de segmentos en total coincide con el total de filas de las matrices. Dichos segmentos se han conectado, junto con las filas, directamente al microcontrolador y los terminales comunes al decodificador. De esta forma se controlan los visualizadores como si fuera otra columna de la pantalla. Este montaje permite ahorrar el uso de un convertidor BCD a 7 segmentos, pero dicha conversión deberá realizarse por *software*.

En la placa de control se usa el mismo microcontrolador que en el proyecto anterior, el PIC 18F452 de Microchip. En este caso la diferencia radica en que sólo se usan entradas digitales y, por tanto, no se usan las entradas analógicas pero sí se usa la característica de las interrupciones externas del microcontrolador.

Respecto a las entradas, existen un total de 9 botones, 3 de ellos para funciones generales de la consola (reiniciar, aumen-



Figura 5: Juego Arkanoid. (Crédito: A. Moser.)

tar velocidad y disminuir velocidad). Los otros 6 se reparten entre los jugadores cuando el videojuego es para dos personas (derecha, izquierda y acción para cada jugador) o conforman un control completo cuando el videojuego es para una sola persona (direcciones y dos acciones).

4.2. Software

El *software* de este proyecto está compuesto por un sistema base y dos videojuegos, uno basado en el *Space Invaders* (matamarcianos) y otro basado en el *Night Driver* (de carreras). Esta primera división en módulos del *software* permite la reutilización del sistema base, junto con el *hardware*, para la programación de otros videojuegos.

Sistema base. El sistema base está compuesto, a su vez, por otros submódulos. Los más importantes son el encargado de la configuración, el encargado de leer las entradas y el encargado de la visualización. Estos dependen de una estructura de datos definida en el mismo sistema base.

Las entradas son leídas cada vez que una de ellas es accionada. Esto se realiza mediante una interrupción, es decir, en el momento de pulsar una entrada el microcontrolador dejará todo lo que esté haciendo para leer las entradas y guardar su estado en una memoria intermedia. Posteriormente, el videojuego que se esté ejecutando en ese momento podrá consultar esta memoria.

Para la visualización se ha optado por mapear la pantalla en memoria reservando un byte entero para cada píxel. De esta forma los videojuegos pueden guardar información sobre el contenido de cada píxel, es decir, no sólo si está encendido o no, sino también si en ese píxel está el jugador, hay un enemigo, hay un obstáculo, etc. Para los visualizadores se reserva un registro donde guardar su valor en código binario. El

módulo de visualización se encarga de adaptar la información guardada en memoria y refrescar la pantalla de forma cíclica mediante una interrupción que se activa por temporización.

Finalmente, el sistema base provee de algunas rutinas básicas a los videojuegos. Estas sirven para la consulta de píxeles, dada su posición absoluta o relativa, y para la modificación de los mismos.

Videojuegos. Con el sistema base en marcha, las acciones que debe ejecutar cada videojuego son:

- *Leer entradas*, usando para ello la memoria intermedia.
- *Tomar decisiones*, consultando para ello el mapa en memoria de la pantalla.
- *Actuar*, reescribiendo en el mapa en memoria de la pantalla.

El primer videojuego a programar está basado en el mítico *Space Invaders*. En este juego, del género “matamarcianos”, el jugador controla una nave provista de una arma para eliminar a las naves enemigas, que también están provistas de armas e intentarán esquivar nuestros disparos. Se puntúa en función del número de enemigos abatidos y la partida acaba cuando el jugador ha sido alcanzado por un proyectil de una nave enemiga un cierto número de veces.

El segundo videojuego a programar está basado en el *Night Driver*. En este juego, del género “de carreras”, el jugador controla un vehículo a través de una sinuosa carretera repleta de obstáculos. La particularidad de este juego está en que, en la pantalla, no es vehículo del jugador el que se mueve, sino la carretera y los obstáculos. Se puntúa según la distancia recorrida y la partida acaba cuando el jugador ha chocado o ha salido de la carretera un determinado número de veces.

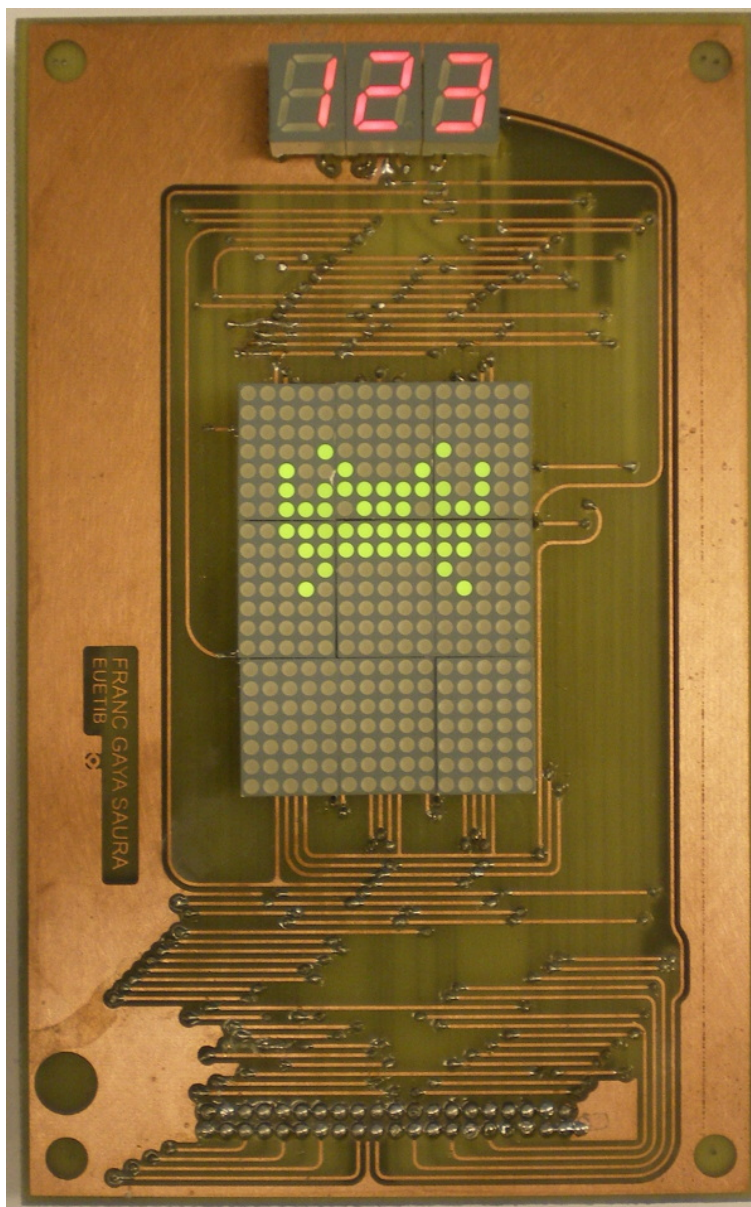


Figura 6: Placa de visualización. (Crédito: F. Gaya).

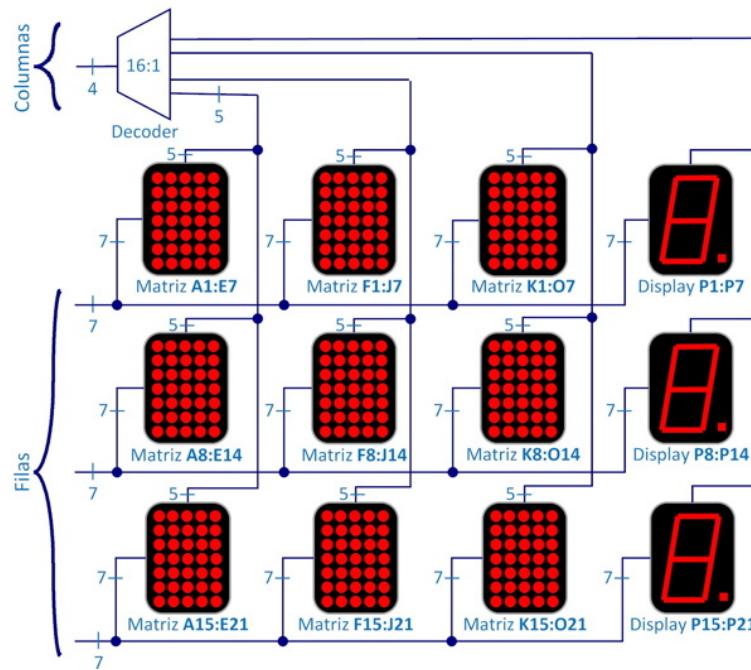


Figura 7: Diagrama de conexión de matrices y visualizadores (Crédito: F. Gaya).

Para más detalles, el lector puede consultar el proyecto detallado, publicado digitalmente con licencia por decisión del autor [4].

5. Desarrollo de una experiencia práctica: videoconsola con carga de cartuchos por USB

Una evolución natural, sugerida por el autor del siguiente proyecto [8], era dotar a la consola de la posibilidad de cargar nuevos programas, simulando las antiguas consolas retro que usaban un sistema de cartuchos.

Para realizar esta simulación se ha optado por un sistema de carga basado en una interficie USB que permite la carga de programas desde el ordenador, mediante un *software* que controla la carga del programa en el microprocesador. Este *software* podrá tener una interficie de usuario que le facilite visualmente la carga del programa. El uso de la interficie USB hace que la videoconsola retro sea fácil de recargar con nuevos programas. Una vez grabado, la videoconsola puede continuar funcionando con el nuevo programa hasta que el usuario decida cambiarlo de nuevo.

La programación de la interficie USB es una técnica que la mayoría de alumnos ven en asignaturas optativas de la titulación de Ingeniería en Electrónica y Automática de nuestro centro. Esta formación previa facilita el desarrollo por parte del alumno de la interficie de carga de nuevos programas.

En la Figura 8 puede verse el prototipo, en estado inicial

de desarrollo en *protoboard*, con la conexión USB y un visualizador gráfico matricial, botones y LED de señalización.

Para este desarrollo se ha optado por un microcontrolador PIC18F4550 de la familia Microchip. El Módulo visualizador LCD es un modelo de la marca Sparkfun Electronics, con resolución de 128×64 píxeles.

El proyecto incorpora un sistema básico de audio, con un altavoz que permite la incorporación de sonidos básicos relacionados con el juego. Debido a la limitación de memoria del microprocesador, no es posible reproducir sonidos digitalizados. Este sería un aspecto a mejorar en futuros desarrollos.

6. Conclusiones

Se ha descrito una propuesta de trabajo para los alumnos de Grado o Máster en Ingeniería Industrial, consistente en un prototipo de videojuego de era retro desarrollado con un microcontrolador específico. Este desarrollo cumple las condiciones que se requieren para un Trabajo Fin de Grado o Máster de aquellas titulaciones con un componente importante en electrónica (diseño de *hardware*) e informática (programación de microcontroladores).

Esta propuesta aúna competencias de programación y de diseño de *hardware* por parte del alumno, aplicadas a un producto final de temática altamente motivadora. Los proyectos aquí referenciados que ya han sido evaluados, han obtenido la calificación de sobresaliente. De la elevada motivación conseguida en los alumnos involucrados, y su realimentación positiva, se desprende que esta aproximación es un buen camino a

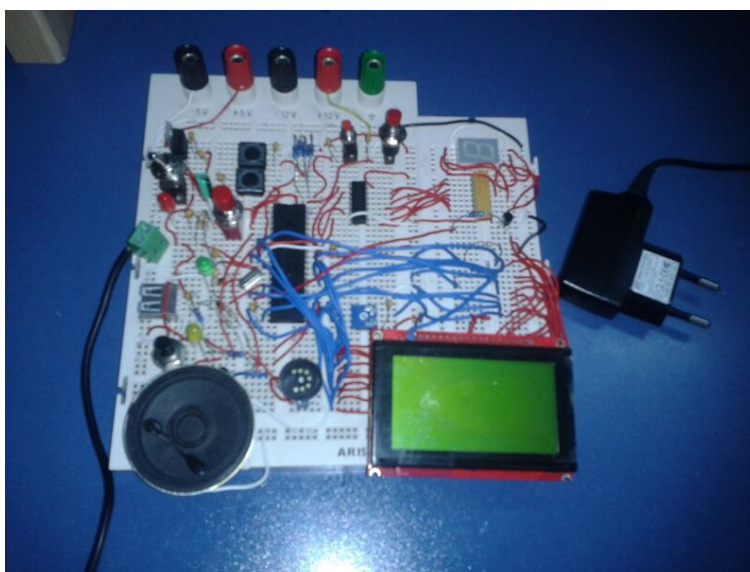


Figura 8: Prototipo de desarrollo (Crédito: R. Ramírez).

seguir. En la parte negativa, algunos alumnos comentan la dificultad de llevar a cabo un desarrollo que involucra en paralelo diseño y fabricación de *hardware*, así como programación en el reducido tiempo de un cuatrimestre lectivo, más las semanas previas de realización del anteproyecto.

En próximos cuatrimestres nos planteamos ampliar el espectro de Trabajos Fin de Grado propuestos con esta temática, e introducir el microcontrolador utilizado en prácticas de otras asignaturas de la intensificación.

Esta aproximación es perfectamente realizable y ampliable por parte de otros alumnos y docentes de nuestras universidades, dado el carácter fácilmente accesible del material requerido para su desarrollo, abriendo puertas a su creatividad.

Agradecimientos

Debemos agradecer sinceramente a los alumnos que han realizado este proyecto como su Trabajo Fin de Grado (referenciados a continuación) por su trabajo, y por sus aportaciones a la concepción inicial de la consola retro. Con su creatividad, ideas y entusiasmo han hecho evolucionar estas miniconsolas hasta convertirlas en un verdadero kit didáctico de desarrollo. Sin sus contribuciones, esta publicación no habría sido posible.

Referencias

- [1] Francisco Charte Ojeda: *Programación en ensamblador*, Ed. Anaya, 2003.
- [2] (sin autor): *Pocket Pong. Un juego antiguo con hardware moderno*. Elektor, ISSN 0211-397X, 291, pp 58-61, 2004 .
- [3] Moti Frank, Ilana Lavy y David Elata: *Implementing the Project Based Learning approach in an academic Engineering course*. International Journal of Technology and Design Education, vol. 13, pp. 273–288, 2003.
- [4] Francesc Gaya: *Programación de videojuegos retro*. Proyecto fin de carrera de Ingeniería Técnica en Electrónica Industrial. EUETIB-UPC, 2010. Consultado el 10/3/2014 en: <http://hdl.handle.net/2099.1/10627>
- [5] Albert López Bull: *Programación con microcontrolador y desarrollo de hardware para la realización de videojuegos de los años ochenta*. Proyecto fin de carrera de Ingeniería Técnica en Electrónica Industrial. EUETIB-UPC, 2010.
- [6] E. Martín Cuenca y I. Angulo Martínez: *Microcontroladores PIC, La solución en un Chip*. Ed. Paraninfo, 2005.
- [7] Alejandro Moser Passini: *Videoconsola con microcontrolador*. Proyecto fin de carrera de Ingeniería Técnica en Electrónica Industrial. EUETIB-UPC, 2009.
- [8] Raúl Ramírez: *Diseño y realización de una videoconsola de videojuegos retro*. Trabajo fin de grado en Electrónica Industrial y Automática. EUETIB-UPC. Pendiente de publicación.
- [9] José María Usategui: *Microcontroladores PIC. Diseño práctico de aplicaciones*. Ed. Paraninfo, 2005.

- [10] Sofía Yep Lee: *Programación de videojuegos retro*. Trabajo fin de grado en Electrónica Industrial y Automática. EUETIB-UPC, 2013. Consultado el 10/3/2014 en: <http://hdl.handle.net/2099.1/18874>



Antoni Perez-Poch es profesor del Departamento de Lenguajes y Sistemas informáticos de la UPC Universitat Politècnica de Catalunya-BarcelonaTech, con docencia en la EUETIB, Escuela de Ingeniería Técnica Industrial de Barcelona. Sus áreas de interés son la investigación educativa, el procesamiento digital de imágenes y la experimentación en microgravedad.

©2014 A. Pérez-Poch. Este artículo es de acceso libre, distribuido bajo los términos de la Licencia Creative Commons de Atribución, que permite copiar, distribuir y comunicar públicamente la obra en cualquier medio, sólido o electrónico, siempre que se acrediten a los autores y fuentes originales.